Table of Contents

[Analysis 2](#_Toc32073325)

[User Needs 2](#_Toc32073326)

[Project Objectives 2](#_Toc32073327)

[Data Dictionary 3](#_Toc32073328)

# Analysis

## User Needs

Rough quotations from voice notes of the user describing expectations of the system.

*“User friendly system that allows user to input contacts... The system should be able to differentiate people based on their occupation, skills, sets, location etc.”*

*“Sections which is based on course… everyone I am directly in contact in… searches that sieve through contacts… e.g. ‘App people’ will show all contacts with ‘app people’ click/hover on a person, comes up with number, email, all their contact details… keeping a record of everyone you know”*

*“Getting people’s contacts, personal type thing… a simple way to put in ‘user’ he does ‘occupation’… when ready, find a dropdown, does ‘occupation’, picks up all the people that does said ‘occupation’”*

## Project Objectives

A record based system, which allows the user to input personal contacts with details about them. The user should be able to interact with a GUI to create, delete and view contacts. A database will store the specific details recorded of a client (‘Firstname’, ‘Lastname’, ‘Phone’, ‘Email’, ‘Occupation’, ‘Location’, ‘Skills’) and will be used to retrieve client data in addition.

Necessary Modules:

* Tkinter
* Messagebox (imported from tkinter, creates messagebox)
* Ttk (imported from tkinter, create combobox, drop downs)
* Sqlite3
* Os
* Font (imported from tkinter.font)
* Msgpanel, seedismisspanel (imported from demopanels)

Final Objectives:

1. Home application, which will have a small list of clients in the database. The user will have access to the following features through the home application:
   1. **New Record:** Creates a window containing fields (e.g. Firstname, Lastname, Occupation) which the user can input into. When the inputs are retrieved, the program will store the details into a database for the user to view.
   2. **Delete Record:** Creates a window which shows all records. The user will be able to select a record to delete from the database.
   3. **View Contacts:** Reads all records from the database. Creates a spreadsheet-like window to show all users from the database. Window has two mini features:
      1. *Sort By,* a dropdown menu containing all the different occupations that are already written. User can select an occupation and the window will update to show all records with that said occupation.
      2. *Find,* user can input a name (either firstname or lastname) to see if a record exists in that database
   4. **Exit:** Close the application
2. When the application is loaded, a small listbox in the home application will be updated to show the records in the database
   1. **Update Listbox:** When the main application is loaded, the listbox within the main window will be updated to show the: Firstname, Lastname, Phone, Email, Occupation and location

## Data Dictionary

Below include all fields that a record will hold when a user creates a new database.

\*Fields which will be used to filter records

|  |  |
| --- | --- |
| FIELD | DATA TYPE |
| Firstname | TEXT |
| Lastname | TEXT |
| Location | TEXT |
| Number | TEXT |
| Email | TEXT |
| \*Field of Work | TEXT |
| \*Souce (linkedin, facebook etc.) | TEXT |
| Occupation | TEXT |

74

# Design

**Stage 1: Features**

Home Application

1. New Record
   * User can input into fields (e.g. Firstname, Lastname, Occupation) to create a new record
   * Once the user’s inputs are retrieved, the record will be written to a database.
2. Delete Record
   * Database will query all records to a window
   * User can select which record to delete from the database
   * Client will be removed from the database
3. View Contacts
   * Database will select all records from the database
   * The records will be displayed in a listbox within a window, containing columns of details (Source, Firstname, Lastname, FieldOfWork, Occupation, Location, Telephone, Email)
   * User can select a sort by dropdown menu to filter depending on occupation
   * Database will only query records that have that occupation
   * Find field where user can enter the firstname or lastname of a record
   * The database will query the record with a firstname or lastname
4. Logout
   * Closes the main application

**Stage 2: System Flow**

